NOMBRE: [redacted]
LIBRETA: [redacted]
CARRERA: COMPUTACIÓN

1	2	3	TOTAL	COND
12	12	10	34	P

Lógica y Computabilidad - 1°C 2023
Recuperatorio de Computabilidad

El parcial dura 5 horas, se necesitan 20 puntos para aprobar y 28 para promocionar. Justifique todos sus razonamientos y cite claramente los resultados teóricos utilizados. Recuerde que $0 \in \mathbb{N}$ y que *computable* significa total y computable.

ENTREGUE LOS EJERCICIOS EN HOJAS SEPARADAS

Ejercicio 1. Sea $\mathsf{esConj} : \mathbb{N} \to \{0,1\}$ un predicado que decide si un número pensado como una lista es un conjunto, es decir, si no tiene elementos repetidos. Así, se tiene:

$$\mathsf{esConj}([10,3,0,5,3,1]) = \mathsf{esConj}([0,0,1,2,3]) = 0\,,$$
$$\mathsf{esConj}([2,10,3,0,5,1]) = \mathsf{esConj}([1,2,3,0,0]) = \mathsf{esConj}([\ \ ]) = 1.$$

(6 p) a) Probar que esConj es primitivo recursivo.

(6 p) b) Probar que también es primitivo recursivo el predicado $\mathsf{mismoConj} : \mathbb{N}^2 \to \{0,1\}$ con la propiedad de que $\mathsf{mismoConj}(m,n)$ es verdadero sii m y n son conjuntos y coinciden como tales, de modo que

$$\mathsf{mismoConj}([1,2,3],[3,1,2]) = \mathsf{mismoConj}([\ \ ],[\ \ ]) = 1\,,$$
$$\mathsf{mismoConj}([1,2,3],[4,1,2]) = \mathsf{mismoConj}([1,1,3],[1,3,1]) = 0.$$

Ejercicio 2. Dados un programa P y un número $x \in \mathbb{N}$ decimos que P *cicla trivialmente con entrada* x si el cómputo $s_0, s_1, s_2, \ldots$ generado por P a partir de la entrada x repite algún snapshot no terminal s_t.

(3 p) a) Demostrar que si P cicla trivialmente con entrada x entonces $\psi_P^{(1)}(x)\uparrow$. Mostrar con un ejemplo que si $\psi_P^{(1)}(x)\uparrow$ no necesariamente se verifica que P cicla trivialmente con entrada x.

(9 p) b) Considerar la función parcial $\mathsf{STOP} : \mathbb{N}^2 \to \mathbb{N}$ dada por

$$\mathsf{STOP}(n,x) = \begin{cases} 1 & , \text{ si } \Phi_n(x)\downarrow; \\ 0 & , \text{ si } n \text{ cicla trivialmente con entrada } x; \\ \uparrow & , \text{ en otro caso.} \end{cases}$$

Demostrar, sin omitir detalles, que STOP es una función parcial computable.

Ejercicio 3. El objetivo de este ejercicio es estudiar la computabilidad del conjunto

$$\mathrm{COMP} = \{\, n \in \mathbb{N} : \mathrm{dom}(\Phi_n) \text{ es un conjunto computable} \,\}.$$

A modo de ejemplo, notar que $n \in \mathrm{COMP}$ si $\mathrm{dom}(\Phi_n)$ es finito y que $n \notin \mathrm{COMP}$ si n es el número de un programa que computa la función parcial $x \mapsto \Phi_x(x)$, pues en tal caso $\mathrm{dom}(\Phi_n) = \mathrm{K}$.

(2 p) a) Demuestre que la función $\mathsf{G} : \mathbb{N}^2 \to \mathbb{N}$ definida como sigue es parcial computable:

$$\mathsf{G}(x,y) = \begin{cases} \Phi_y(y) & , \text{ si } \Phi_x(z)\downarrow \text{ para todo } z \le y; \\ \uparrow & , \text{ en otro caso.} \end{cases}$$

(4 p) b) Para cada $a \in \mathbb{N}$ sea $\mathsf{g}_a : \mathbb{N} \to \mathbb{N}$ dada por $\mathsf{g}_a(y) = \mathsf{G}(a,y)$. Demuestre las equivalencias:

$$a \notin \mathrm{TOT} \iff \mathrm{dom}(\mathsf{g}_a) \text{ es finito} \iff \mathrm{dom}(\mathsf{g}_a) \text{ es computable.}$$

(6 p) c) Decida si COMP es p.r., computable, c.e. o co-c.e.

6|6 [redacted]

[redacted]

30/06/23

1 (a) $esconj(l)$ es pr. Demostración:

Lo pruebo por composición de funciones que son p.r. Creo primero unas funciones p.r. aux:

- $ap(l,m)$ tal que $ap: \mathbb{N}^2 \to \mathbb{N}$ nos indica la cantidad de apariciones de m en l (visto como una lista).

  $ap(l,m) := \sum_{i=1}^{\overset{\to pr}{|l|}} (\underset{\to pr}{l[i]} \underset{\to pr}{=} \underset{\to pr}{m})$ es pr por composición de pr's.

- $único(l,m)$ tal que $único: \mathbb{N}^2 \to \{0,1\}$ es un predicado que nos indica si m aparece una sola vez en l (lista)

  $único(l,m) := \underbrace{ap(l,m)}_{\text{pr demostrado antes}} \overset{\to pr}{=} 1^{\to pr}$ es pr por composición

Entonces, con las funciones auxiliares armo esconj

$esconj(l) := \prod_{i=1}^{\overset{\to pr}{|l|}} \underbrace{único(l, l[i])}_{\text{pr demostrado antes}}$.

pr

$\Rightarrow$ esconj la puedo armar por composición de funciones pr y por lo tanto es pr también. $\square$

Obs: Vemos que si algún elemento no fuera único, la productoria daría 0.

✓

(b) mismoConj (c, d) es p.r. Demostración:
Con la misma idea de antes voy a hacer composición de ~~p.r.~~ funciones p.r. → pr por (a) pr

$$\text{mismoConj}(c,d) := \text{esConj}(c) \cdot \text{esConj}(d) \cdot \text{mismoLargo}(c,d) \cdot \prod_{i=1}^{\overset{\to pr}{|c|}} \left[\overset{\to pr}{ap(c, c[i])} = \underline{ap(d, c[i])} \right] (*)$$

p.r. ← ⊙ ; pr por (a)

$\text{mismoLargo}: \mathbb{N}^2 \to \{0,1\}$ es pr por composición.

$$\text{mismoLargo}(c,d) := \underbrace{|c|}_{pr} = \underbrace{|d|}_{pr} \Rightarrow pr$$

(por composición)

(*) esta productoria es pr y verifica que todos los elementos de c estén en d, y como ambos son conjuntos (por esConj) y tienen la misma longitud, entonces debe ser el mismo conjunto.

□

55

✓

a	b
3	9

[redacted]

2/3
30/06/23

$\boxed{2}$ a) Si P cicla trivialmente con entrada x significa que ~~tengo~~ existen i, j tal que $i<j$ pero $s_i = s_j$ (con s_i no terminal). Entonces, si esto pasa (que llego al mismo estado de cómputo con s_j que ya tuve en el "pasado" con s_i) ~~voy~~ el programa P se va a comportar como lo hizo en el tiempo i. ~~Si~~ miramos esto continuamente, vamos a ver que si el s_i nos llevó al s_j después de $j-i$ pasos, entonces nos volverá a llevar a un s_k con $k>j$ tal que $s_k = s_j = s_i$, y así sucesivamente ~~...~~ de forma infinita.
Por lo tanto $\Psi_P(x)\uparrow$ si P cicla trivialmente con entrada x.

Veamos un ejemplo que al revés no ocurre.
Tengo el programa P:

```
[L] Z1 ← Z1 + 1
    IF Z1 ≠ 0 GOTO L
```

P se indefine porque siempre $Z_1 \neq 0$ ya que empieza en 1 y luego crece.

Vemos que siempre que pase por la primera instrucción, Z_1 va a ser diferente ya que cambia constantemente con esa instrucción, en sí y en la segunda también siempre vamos a tener un Z_1 diferente a cualquier estado anterior.

(b) $STOP(M,X) = \begin{cases} 1 & \text{(1) } \Phi_M(x)\downarrow \\ 0 & \text{(2) } M \text{ cicla trivialmente con entrada } x \\ \uparrow & \text{(3) cc} \end{cases}$

Si puedo construir un programa en S que lo compute entonces STOP es parcial computable. Hago un pseudo-programa STOP:

$$
\begin{array}{ll}
[I] & Z_2 \leftarrow Z_1 \\
 & Z_1 \leftarrow Z_1 + 1 \\
 & \text{IF } STP(X_1, X_2, Z_1) \neq 0 \text{ GOTO } F \\
 & S_1 \leftarrow SNAP(X_1, X_2, Z_1) \\
[R] & S_2 \leftarrow SNAP(X_1, X_2, Z_2) \\
 & \text{IF } R(S_1, S_2) \neq 0 \text{ GOTO } E \\
 & \text{IF } Z_2 = 0 \text{ GOTO } I \\
 & Z_2 \leftarrow Z_2 \dot{-} 1 \\
 & \text{GOTO } R \\
[F] & Y \leftarrow Y + 1
\end{array}
$$

STP es p.r., así que lo puedo usar como predicado, y verifico la primera (1) condición.

OK! YA ENTENDÍ

tomo a R como el predicado tq $R(x,y) = (x = y)$ que es p.r.
Esto va a implicar que ocurre la segunda (2) condición (i.e., hay 2 estados $s_i = s_j$ con $i \neq j$)

Para el caso de (3) vemos que Z_1 (que representa a los t pasos de STP y SNAP) va a ir creciendo, y se va a seguir verificando si termina o no, por lo tanto, en caso de que no termine y no tenga una repetición de estados (2), se va a colgar probando incrementando Z_1.
Por lo tanto, STOP es parcial computable, ya que pude armar un programa que lo computa.

□

[redacted]

3/3
30/06/23

a	b	c	10
2	2	6	

3 (a) $G(x,y) = \begin{cases} \Phi_y(y) & (1) \text{ si } \Phi_x(z)\downarrow \ \forall z \leq y \\ \uparrow & (2) \text{ cc} \end{cases}$

Construyo un pseudo-programa en S para computar a $G(x,y)$:

```
     Z2 <- X2
[R]  IF STP(Z2, X1, Z1) ≠ 0 GOTO E
     Z1 <- Z1 + 1
     GOTO R
[E]  IF Z2 = 0 GOTO F
     Z2 <- Z2 ∸ 1
     GOTO R
[F]  Y <- Φ(X2, X2)
```

STP es pr, y puedo verificar si el x-ésimo programa termina con entrada z en (1).

Φ es parcial computable, entonces lo puedo anexar al final de mi programa para computar $\Phi_y(y)$.

Para el caso (2) en caso de $\Phi_y(y)$ se indefina al final de todo, o previamente vemos que el x-ésimo programa con alguna de las $z \leq y$ no termina infinitamente, entonces se cuelga. Por lo tanto $G(x,y)$ es parcial computable, porque tiene un programa que lo computa.

(b) $g_a(y) = G(a,y)$. Demostrar:

$a \notin TOT \overset{(I)}{\iff} dom(g_a)$ finito $\overset{(II)}{\iff} dom(g_a)$ comp.

(I) Si $a \notin TOT$ implica que a partir de cierto y $g_a(y)\uparrow$ por la particularidad que $G(x,y)$ tiene. $\Phi_a(y)\uparrow \Rightarrow \forall y' \geq y \ \Phi_a(y')\downarrow$ porque tiene que pasar por el y que se indefine.

↳

En caso que se indefina por el $\Phi_y(y)$, de todos formas va a existir un $y' > y$ a partir del cual se indefinan porque, a no ~~pertenezca~~ computa un programa total.
Entonces, al estar acotado por algún y, sabemos que el dom(ga) es finito.
Al revés pasa que si el dom(ga) es finito, significa que a partir de algún y ~~[illegible]~~ ~~por lo tanto~~ $\Phi_a(y)\uparrow$ (y por lo tanto $g_a(y)\uparrow$), entonces $a \notin TOT$, porque si no $\Phi_a(y)\downarrow$. □

(II) La ida es evidente, porque si ~~es finito~~ dom(ga) va a poder ser computable. La vuelta la puedo ver por

(c) $COMP = \{n : dom(\Phi_n)$ es un conjunto computable$\}$
Intuyo que COMP no es ce ni co-ce, pero por (b) sospecho que no lo puedo reducir a TOT.
Intento ver que $TOT \leq \overline{COMP}$ usando ~~la~~ la función $G(a,y)$ del (a) que ya sé que es pc. $\Rightarrow \exists e \,/\, \Phi_e(a,y) = G(a,y)$
Por Teorema del Parámetro $\forall a \in \mathbb{N}, \exists \underset{\text{pr.}}{S_1^1}(a,e) \,/\, \Phi_{S(a,e)}(y) = \Phi_e(a,y)$
- Si $x = a \in TOT$ entonces $\Phi_a(y)\downarrow \forall y$ y por lo tanto dom(ga) es infinito ($\Phi_y(y)$ se indefine para infinitos casos, ej: para todos los y que representen la id) $\Rightarrow$ por (b) (contrarecíproco) $S_1^1(a,e) \in \overline{COMP}$
- Si $x = a \notin TOT$ entonces $\Phi_a(y)\uparrow$ a partir de algún y (por (b), ya que dom(ga) quedaría finito) y $g_a(y)$ también $\Rightarrow$ por (b) $S_1^1(a,e) \in COMP$

Con esta reducción y sabiendo que TOT no es nada, podemos concluir que $\overline{COMP}$ y COMP no es ~~computable~~ ~~[illegible]~~ ce, co-ce, ni por lo tanto comp o pr. □